**Формулировка задания:**

Решить уравнения - трансцендентное и полиномиальное - методами половинного деления и простых итераций, и исследовать зависимость числа итераций от точности и начального приближения

**Алгоритмы**:

**Метод половинного деления:**

Метод половинного деления годится только для отыскания корней кратности 1. В том случае если корней на отрезке больше одного, то в зависимости от характера функции на входном отрезке данный метод найдет только один корень.

Шаг 1: проверить выполнения требуемых условий, если не выполняются перейти к шагу 6

Шаг 2: проверить длину отрезка, если она меньше удвоенной точности перейти к шагу 5

Шаг 3: вычислить середину отрезка, проверить значение функции в середине отрезка и заменить один из концов отрезка серединой, тот который имеет знак значения функции такой же как и у значение в середине. Если значение функции в середине отрезка равно 0 перейти к шагу 5

Шаг 4: перейти к шагу 2

Шаг 5: ответ - середина отрезка

Шаг 6: конец

**Метод простых итераций:**

Чтобы найти корень уравнения на некотором(конкретном) промежутке этим методом достаточно найти такую функцию(ƒ(х) = х), что ее область определения содержится в данном промежутке и ее производная ограничена по модулю единицей.

Шаг 1: привести уравнение f(x) = 0 к виду x = φ(x)

Шаг 2: проверить выполнение условий, если не выполнены перейти к шагу 1 и попробовать привести другим способом к этому виду, если не получается добиться выполнения условий перейти к шагу 7

Шаг 3: выбрать начальное приближение из подходящего вышеописанным условиям отрезка

Шаг 4: подставить в уравнение x = φ(x), получить следующее приближение

Шаг 5: проверить, что разность следующего и предыдущего приближений больше abs((1 - q)/q) \* ε и перейти к шагу 4 иначе перейти к шагу 6

Шаг 6: последнее приближение - корень f(x) = 0 на данном отрезке, найденный с данной точностью

Шаг 7: конец

**Анализ задачи + проверка условий:**

Предпошлем рассмотрению уравнений теорему о локализации корней:

если функция f(x) непрерывна на отрезке [a, b] и принимает на концах отрезка значения разных знаков, а производная ее сохраняет постоянный знак внутри отрезка, то внутри отрезка существует корень уравнения f(x) = 0 и притом единственный.

**Полином:**

Найдем границы действительных корней полинома:

Применим теорему о верхней границе положительных корней:

эта теорема говорит, что действительные корни находятся в некотором отрезке [min, max], где min и max находятся по правилам(их можно найти, например, на <http://www.abakbot.ru/online-16/274-limit-root-polynom>)

3x^4 + 4x^3 -12x^2 - 5 = 0

max = 1 + sqrt(4) = 3

min = -(1 + 4) = -5

Таким образом положительные корни заключены в отрезке [-5; 3]

Выберем отрезок удовлетворяющий теореме о локализации корней:

x ∈ [1.5; 1.7], на этом отрезке производная монотонно возрастает и значения полинома на концах отрезка разных знаков. Таким образом корень на этом отрезке единственный

**Трансцендентное:**

Выберем отрезок удовлетворяющий теореме о локализации корней:

x ∈ [0.18; 0.196], на этом отрезке производная монотонно возрастает и значения функции на концах отрезка разных знаков. Таким образом корень на этом отрезке единственный

Для МПИ введем общий случай приведения данного уравнения к нужному виду:

x = x - αf(x)

abs(1 - αf ’(x)) < q(α)

q - ?

コ 0 < m <= f ’(x) <= M, x ∈ [a, b] => q = (M - m) / (M + m)

подробнее можно найти в книге Амосов А.А., Дубинский Ю.А., Копченова Н.В. Вычислительные методы для инженеров Глава 4 §4.4.6

**МПД:**

Все условия проверены выше

**МПИ:**

**Полином:**

Дана зависимость: y(x) = 3x^4+4x^3-12\*x^2-5

Найдена зависимость: φ(x) = ((12x^2+5)/(3x^2+4x))^(1/2)

Проверка условий:

φ(1.5)’ ≈ 0.083

φ(1.7)’ ≈ 0.088

φ(1.5) ≈ 1.58

φ(1.7) ≈ 1.6

**Транцендентное:**

Дана зависимость: y(x) = log(x) + (x+1)^3

Найдена зависимость: φ(x) = x - 0.112 \* (log(x) + (x+1)^3)

Проверка условий:

φ(0.18)’ ≈ 0.052

φ(0.196)’ ≈ 0.091

φ(0.18) ≈ 0.189

φ(0.196) ≈ 0.187

**Тестовый пример:**

**Полином:**

дана зависимость: y(x) = 3x^4+4x^3-12\*x^2-5

найти корень на отрезке ｘ ∈ [1.5, 1.7] с точностью ε = 0.001

q = 0.1

**МПД:**

первая итерация:

Шаг 2: 1.7 - 1.5 > 0.002

Шаг 3: middle = 1.6;

вторая итерация:

Шаг 2: 1.6 - 1.5 > 0.002

Шаг 3: middle = 1.55

третья итерация:

Шаг 2: 1.6 - 1.5 > 0.002

Шаг 3: middle = 1.55

четвертая итерация:

Шаг 2: 1.6 - 1.55 > 0.002

Шаг 3: middle = 1.575

пятая итерация:

Шаг 2: 1.6 - 1.575 > 0.002

Шаг 3: middle = 1.5875

шестая итерация:

Шаг 2: 1.6 - 1.5875 > 0.002

Шаг 3: middle = 1.5938

седьмая итерация:

Шаг 2: 1.5938 - 1.5875 > 0.002

Шаг 3: middle = 1.5938

восьмая итерация:

Шаг 2: 1.5938 - 1.5875 > 0.002

Шаг 3: middle = 1.5906

девятая итерация:

Шаг 2: 1.5938 - 1.5906 > 0.002

Шаг 3: middle = 1.5922

десятая итерация:

Шаг 2: 1.5938 - 1.5922 < 0.002

Шаг 5: x = 1.5914

Шаг 6: конец

**МПИ:**

первая итерация:

первые 3 шага выполнены

Шаг 4: x1 = 1.5842

Шаг 5: 1.5842 - 1.5 > 0.009

вторая итерация:

Шаг 4: x2 = 1.5914

Шаг 5: 1.5914 - 1.5842 < 0.009

Шаг 6: x = 1.5914

Шаг 7: конец

**Транцендентное:**

Дана зависимость: y(x) = log(x) + (x+1)^3

Найти корень на отрезке x ∈ [0.18, 0.196] с точностью ε = 0.001

q = 0.1

**МПД:**

первая итерация:

Шаг 2: 0.196 - 0.18 > 0.002

Шаг 3: middle = 0.188;

вторая итерация:

Шаг 2: 0.188 - 0.18 > 0.002

Шаг 3: middle = 0.184

третья итерация:

Шаг 2: 0.188 - 0.184 > 0.002

Шаг 3: middle = 0.186

четвертая итерация:

Шаг 2: 0.188 - 0.186 > 0.002

Шаг 3: middle = 0.187

пятая итерация:

Шаг 2: 0.188 - 0.187 < 0.002

Шаг 5: x = 0.1875

Шаг 6: конец

**МПИ:**

первая итерация:

первые 3 шага выполнены

Шаг 4: x1 = 0.188

Шаг 5: 0.188 - 0.18 < 0.009

Шаг 6: x = 0.188

Шаг 7: конец

**Перечень контрольных тестов:**

**МПД:**

1. y(x) = 3x^4+4x^3-12\*x^2-5; x ∈ [1.5, 1.7]; ε = 0.1; x = 1.6
2. y(x) = 3x^4+4x^3-12\*x^2-5; x ∈ [1.5, 1.7]; ε = 0.0001; x = 1.5921
3. y(x) = log(x) + (x+1)^3; x ∈ [0.18, 0.196]; ε = 0.001; x = 0.1875
4. y(x) = log(x) + (x+1)^3; x ∈ [0.18, 0.196]; ε = 0.0001; x = 0.1874

**МПИ:**

1. φ(x) = ((12x^2+5)/(3x^2+4x))^(1/2); x ∈ [1.5, 1.7]; ε = 0.01; x = 1.5920; q = 0.1
2. φ(x) = ((12x^2+5)/(3x^2+4x))^(1/2); x ∈ [1.5, 1.7]; ε = 0.0001; x = 1.5921; q = 0.1
3. φ(x) = x - 0.112 \* (log(x) + (x+1)^3); x ∈ [0.18, 0.196]; ε = 0.001; x = 0.1875; q = 0.1
4. φ(x) = x - 0.112 \* (log(x) + (x+1)^3); x ∈ [0.18, 0.196]; ε = 0.0001; x = 0.1874; q = 0.1

**Модульная структура программы:**

**Программа состоит из одного модуля с пятью вложенными функциями:**

В области видимости всех функций находится массив из пяти элементов: границы отрезка, количество итераций, точность, корень

function half\_division\_method (a, b, d, func);

Ищет корень методом половинного деления уравнения func с точностью d на [a, b]

function simple\_iterations\_method(x0, а, d, q, func\_1, func\_2);

Ищет корень методом простых итераций уравнения func\_1 с точностью заданной функцией func\_2 с точностью d на [x0, a], где х0 - начальное приближение и где q - число, ограничивающее производную func\_1

function k\_on\_d(a, b, d, q, func\_1, func\_2, method\_name, x\_label, y\_label, function\_);

Выводит зависимость количества итераций от точности, где a - левая, b - правая границы отрезка, на котором ищется корень уравнения func\_1(эти два параметра необходимо передавать, тк их надо передавать в другие аргументы данной функции), d - точность, q - число, ограничивающее производную func\_1

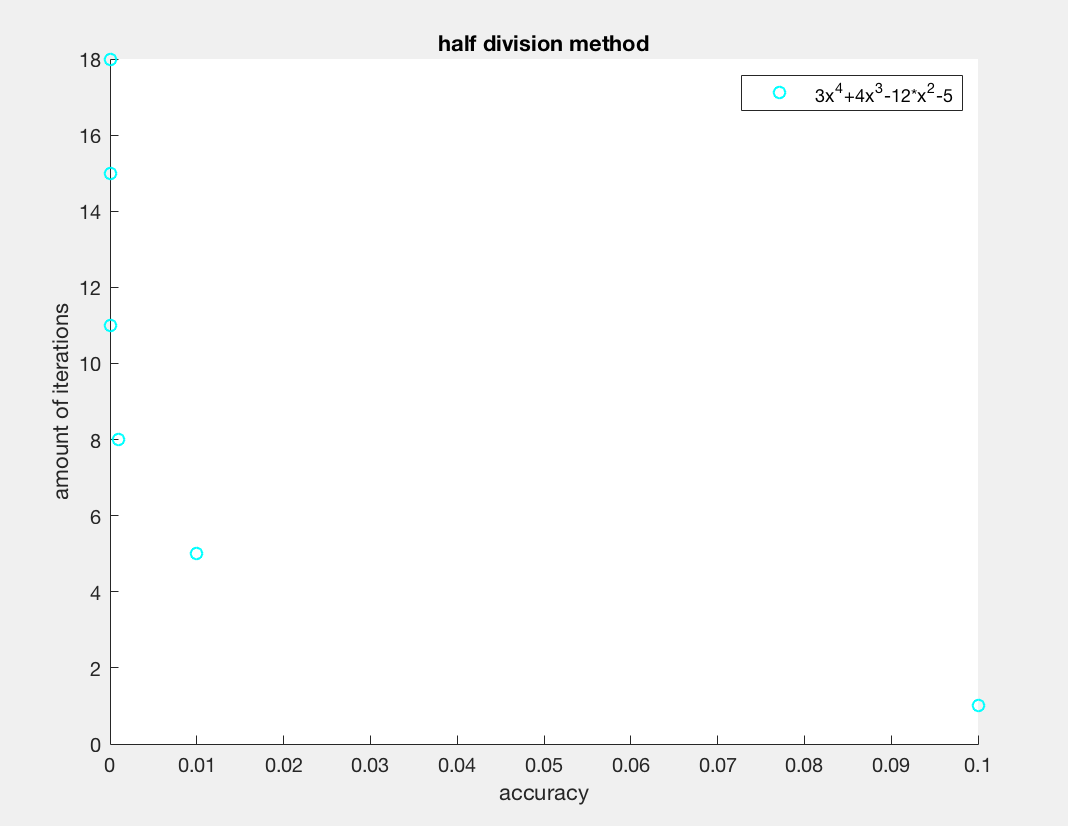
function k\_on\_x0(a, b, d, q, func\_1, func\_2, method\_name, x\_label, y\_label, function\_);

Выводит зависимость количества итераций от начального приближения, где a - левая, b - правая границы отрезка, на котором ищется корень уравнения func\_1(эти два параметра необходимо передавать, тк их надо передавать в другие аргументы данной функции), d - точность, q - число, ограничивающее производную func\_1

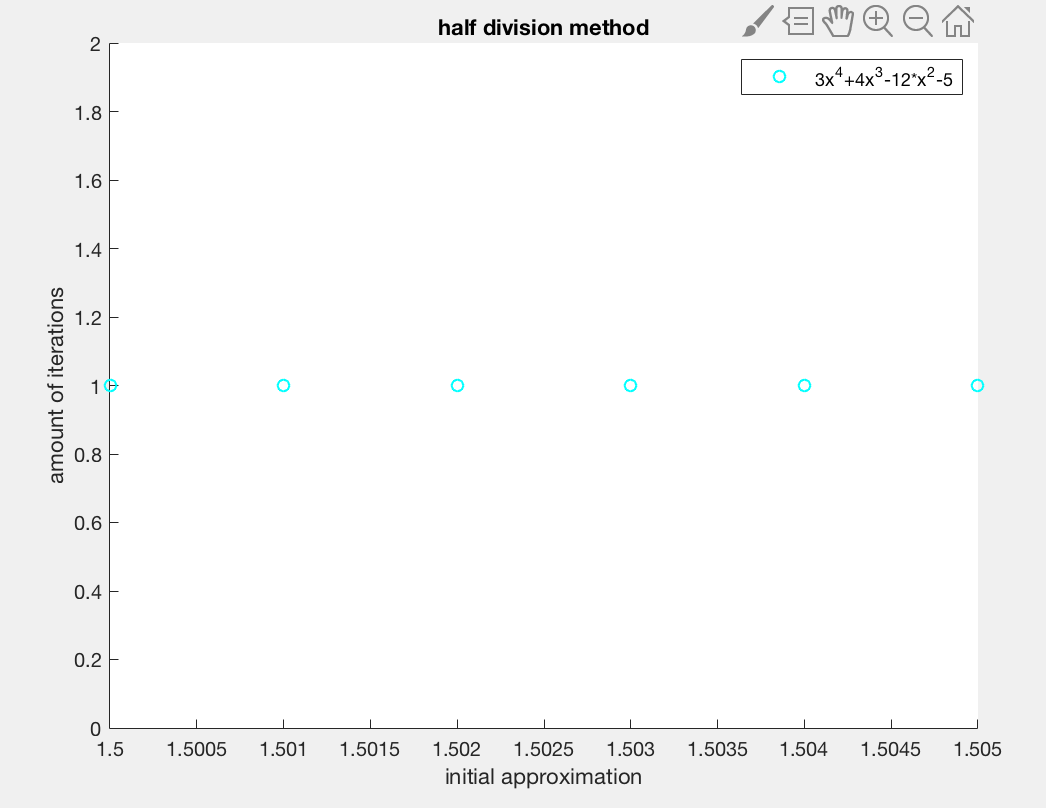
function convergence(a, b, func, method\_name, dependence);

Наглядно(на графике) показывает характер сходимости последовательности точек полученных из func(корень, которой необходимо найти), при помощи метода(method\_name) к корню, на [a, b], где dependence - строка, содержащая строку с зависимостью

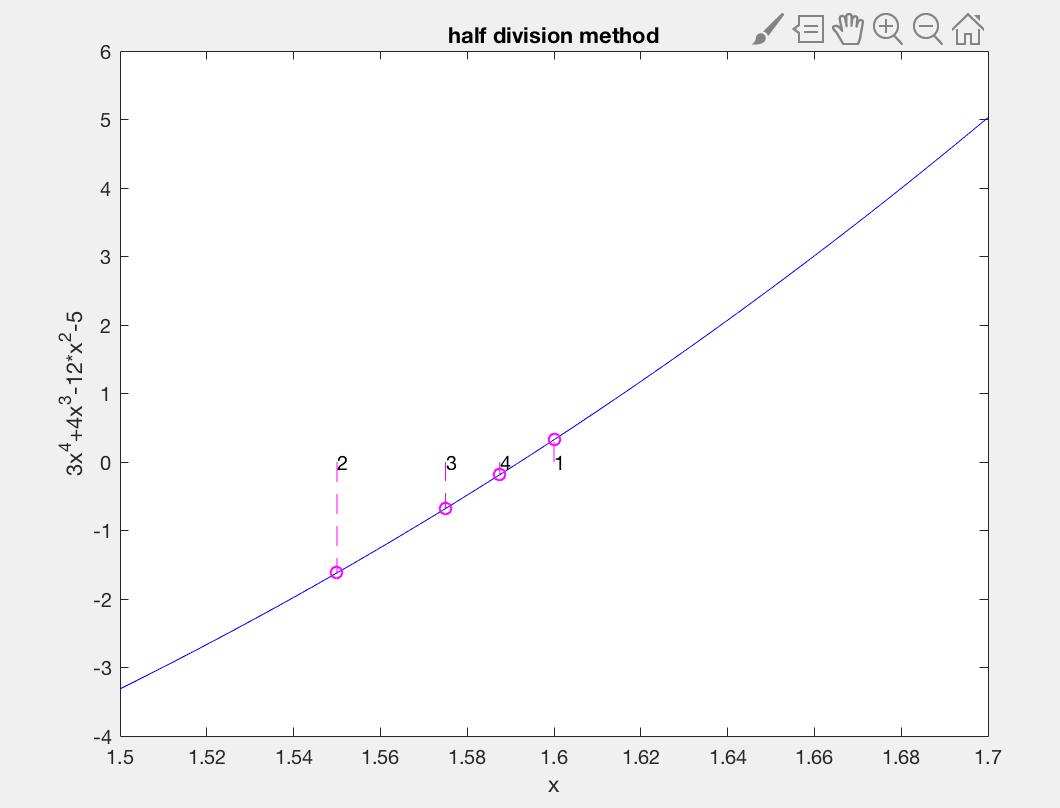
**Численный анализ решения:**



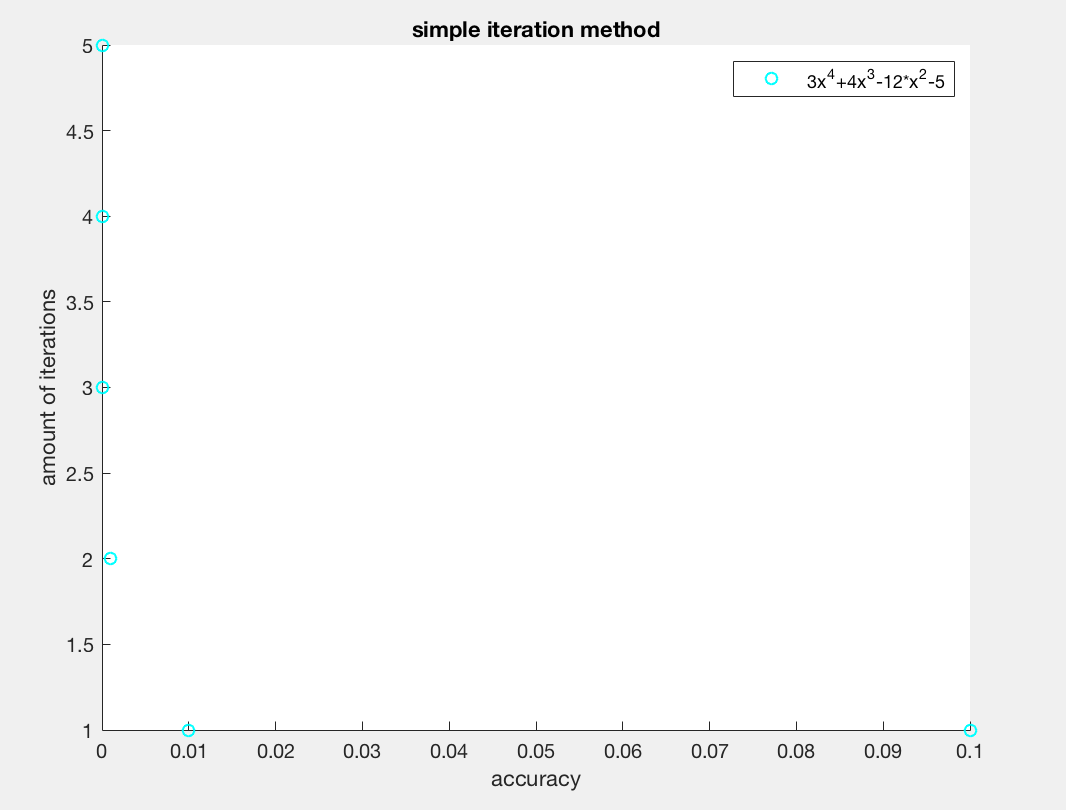
На этом графике можно видеть экспоненциальную зависимость количества итераций от точности



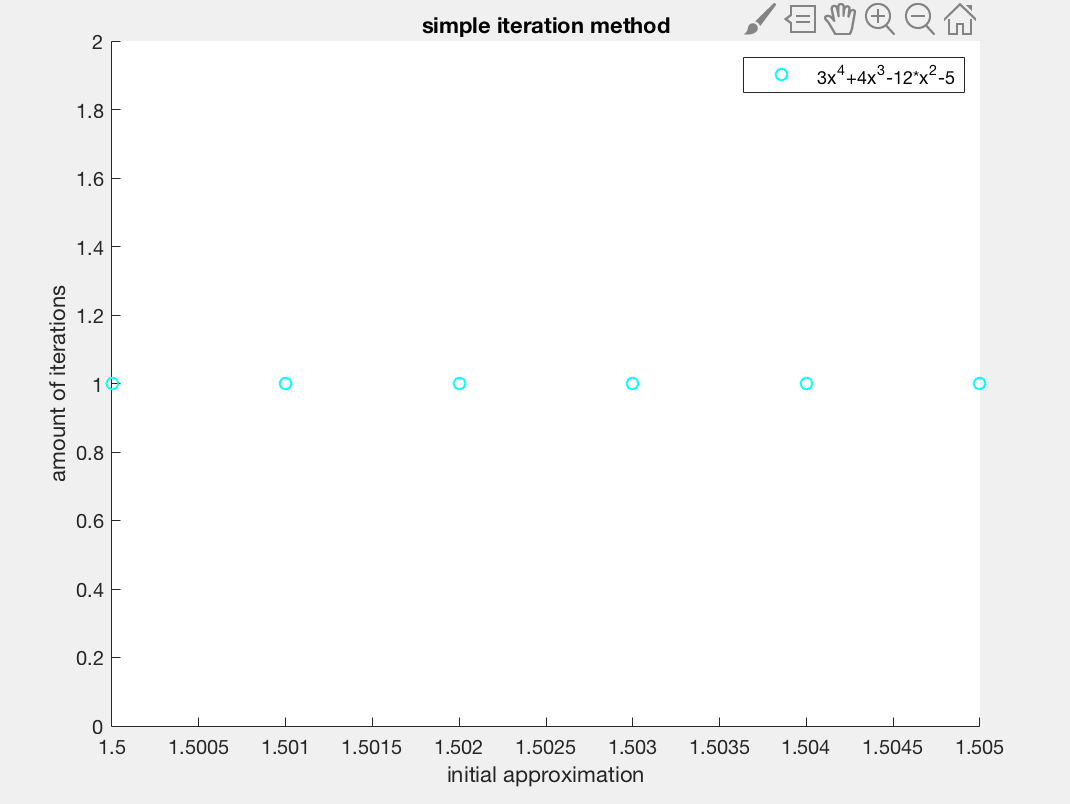
Такая зависимость получилась из-за шага - слишком маленький, чтобы что-то поменялось заметно. Наихудший расклад для этого метода - один из краев входного отрезка очень близок к корню, наилучший - корень очень близок к середине входного отрезка



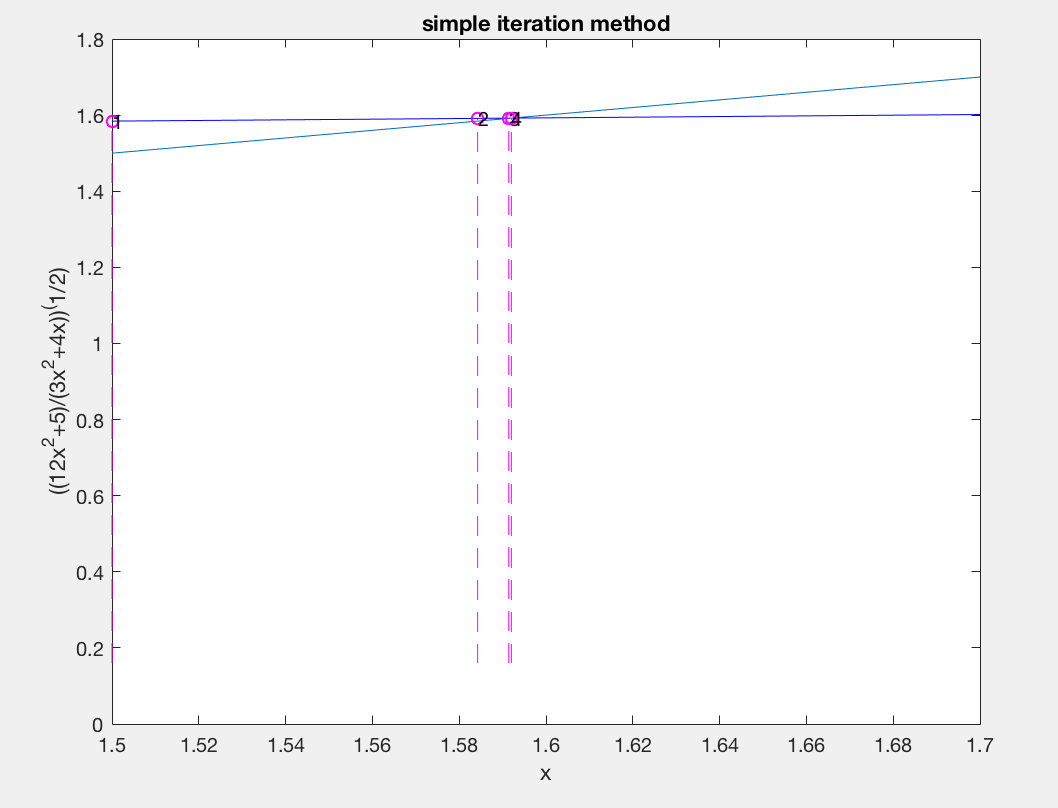
На этом графике наглядно видно сходимость к корню



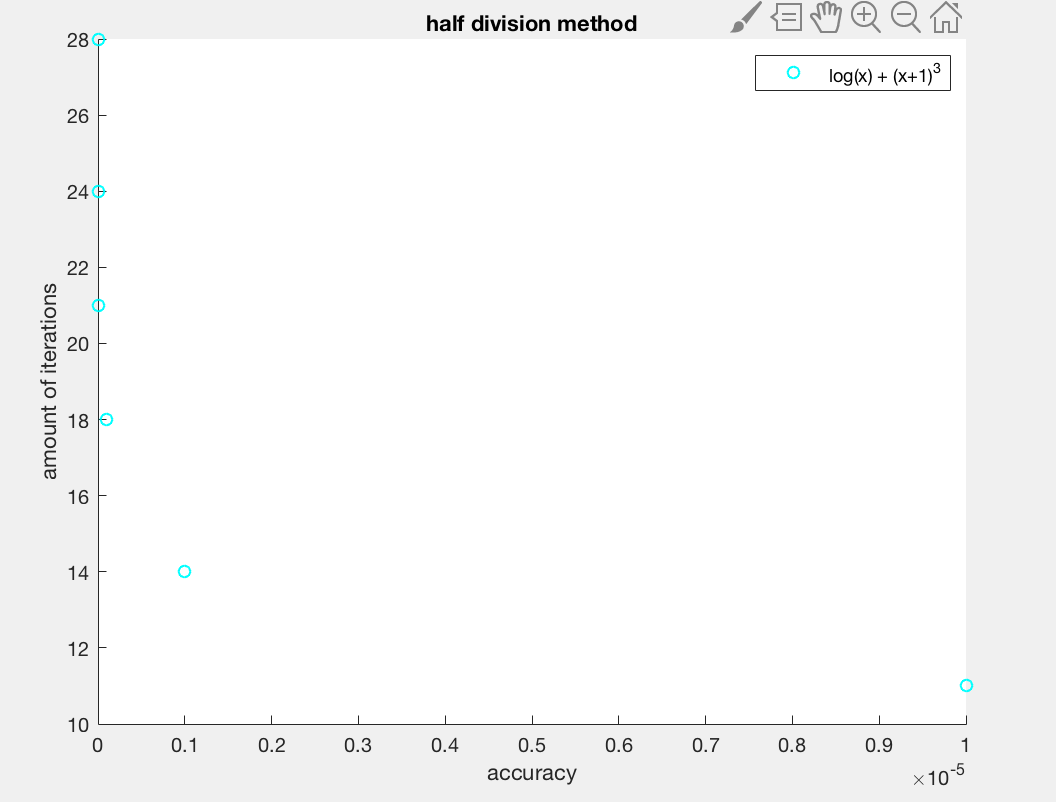
Опять же экспоненциальная зависимость



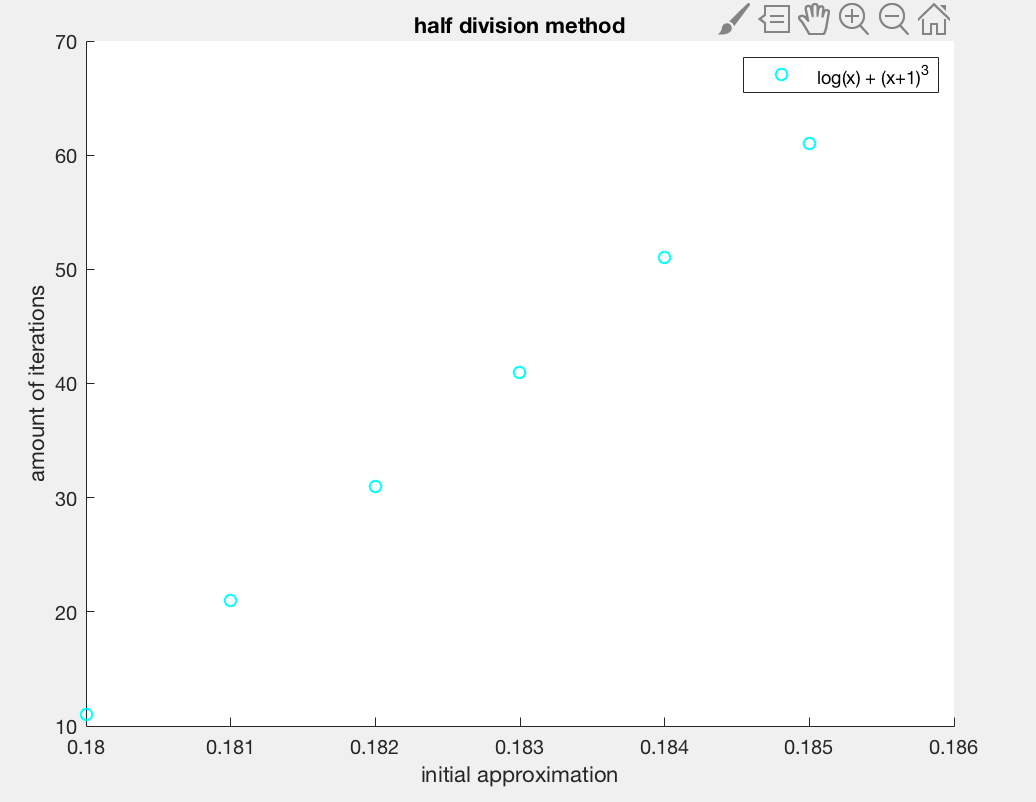
Для метода простых итераций наилучший расклад - корень близко к начальному приближению. Такая зависимость - малость шага



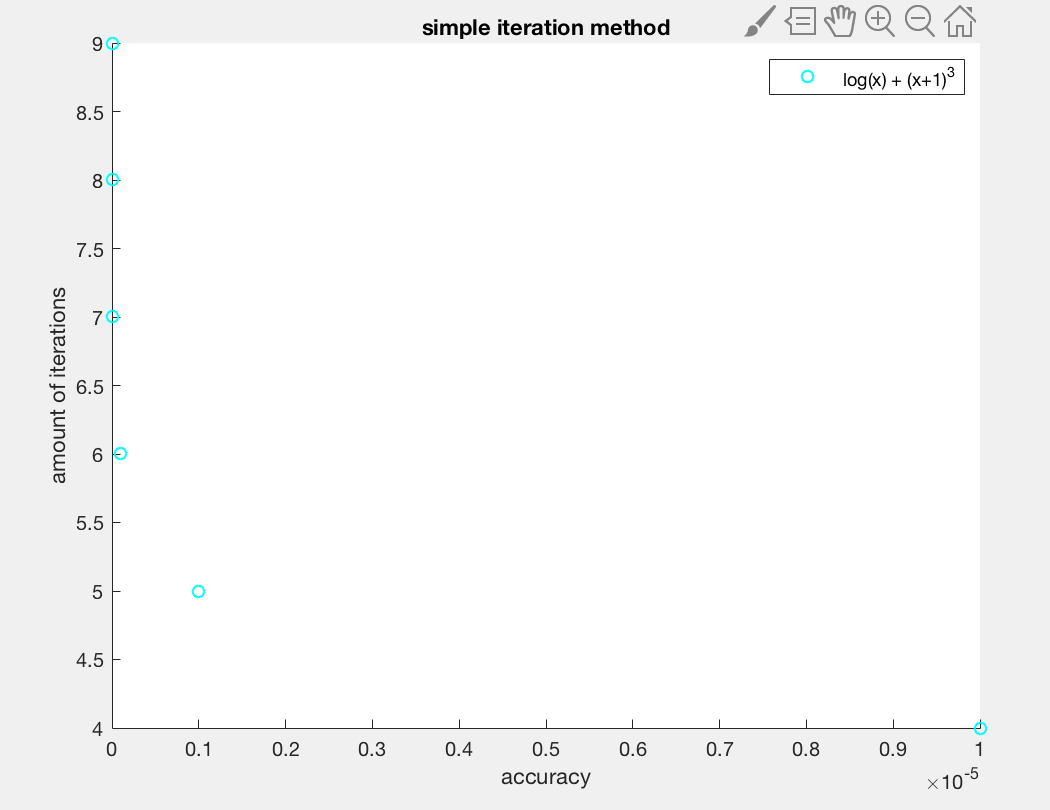
Можно, приглядевшись, увидеть характер сходимости



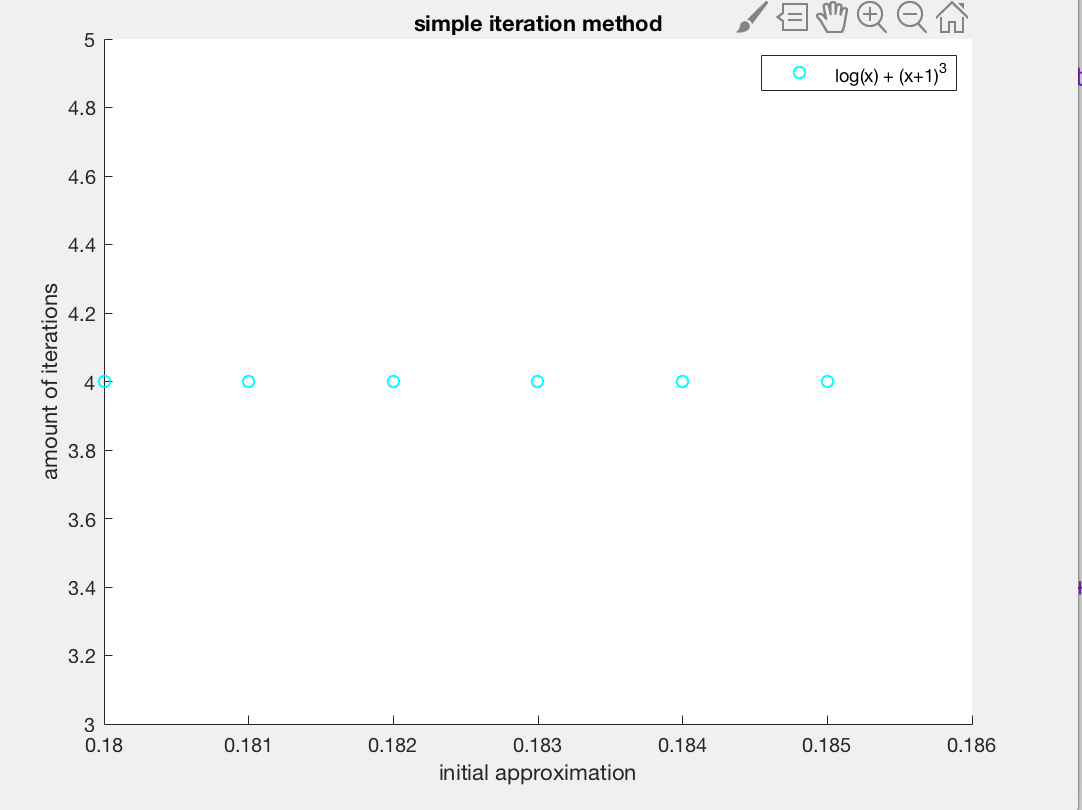
Экспоненциальная зависимость



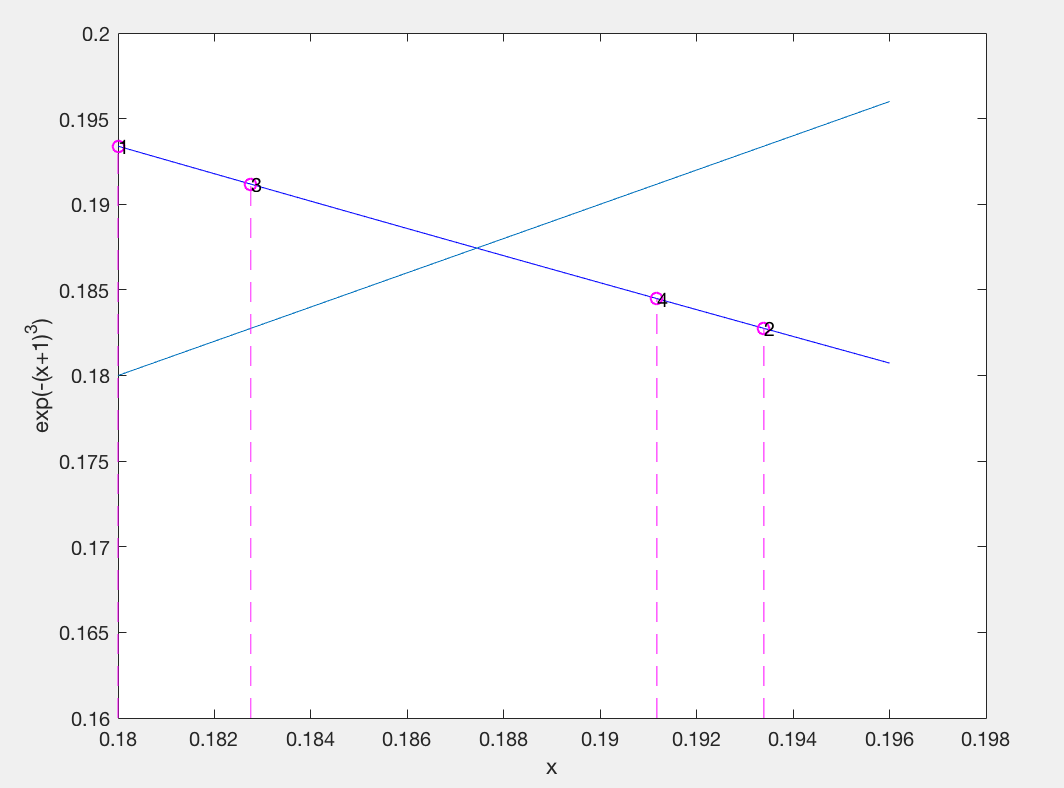
Вот теперь видно то увеличение количества итераций(линейное) от начального приближения



Экспонента



Из-за того, что интервал хорошо подобран изменение в количестве итераций не происходит(слишком узкий отрезок и начальное приближение изначально достаточно близко к корню)



Характер сходимости можно увидеть на графике выше - спираль

**Вывод:**

МПД - прост и интуитивно понятен

МПИ - чуть сложнее в реализации

Оба метода сходятся со скоростью геометрической прогрессии: МПД с q = ½, МПИ с некоторым q, который довольно легко подобрать меньше ½

На современных компьютерах разница не заметна, поэтому, если нет острой необходимости, проще использовать ПМД(хотя я уверен есть пакеты в которых это уже давно сделано и настолько же давно протестировано и оптимизировано)